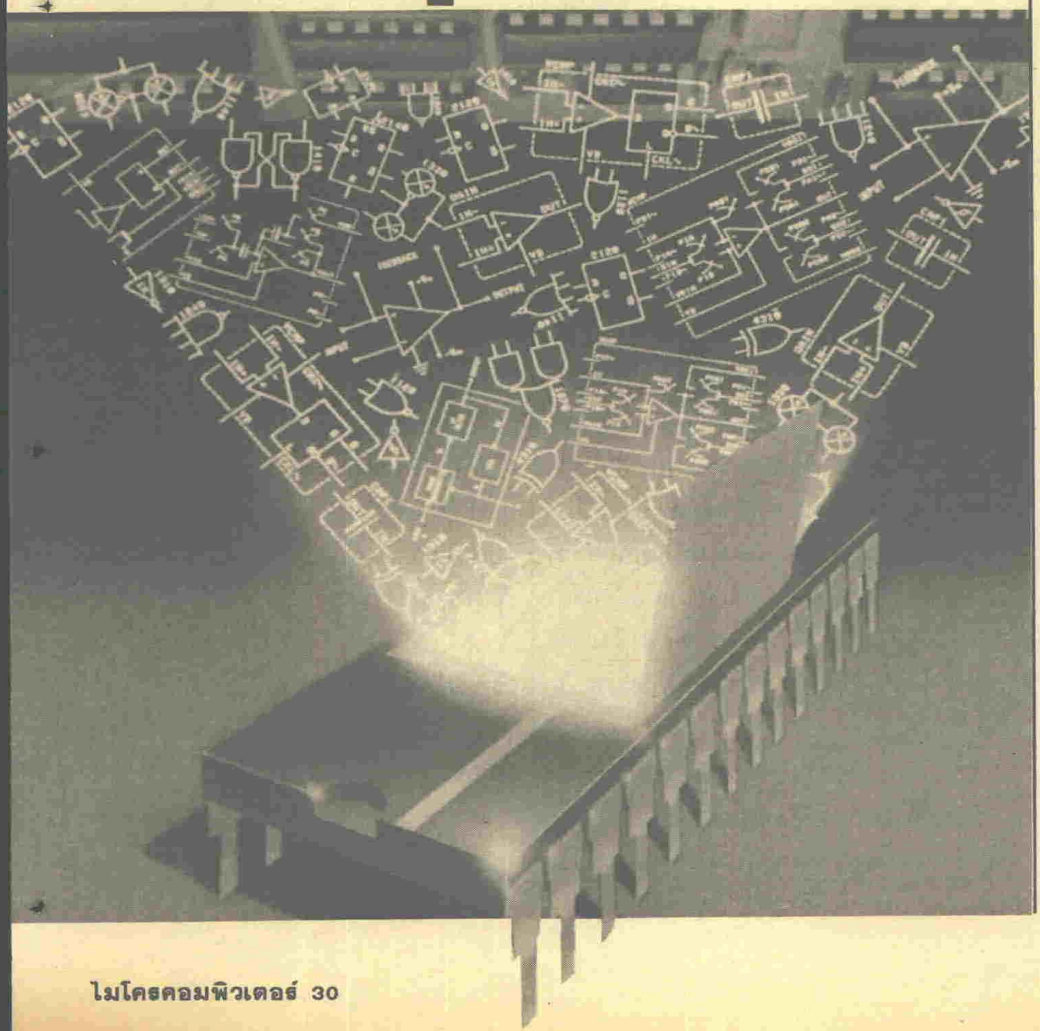
บุญเลิศ เอี่ยมทัศนา

ในวงการไมโครคอมพิวเตอร์ยุคสมัยของเครื่อง 8 บิต กำลังจะผ่านไปยุคสมัยของเครื่อง 16 บิต และอาจจะเลยไปถึง 32 บิต กำลังจะเข้ามาแทนที่ มีผู้ตั้งข้อสงสัยว่าเมื่อเป็นเช่นนี้ทำไมการเรียนการสอนทางด้านฮาร์ดแวร์จึงยังคงใช้ซีพียูขนาด 8 บิตอยู่

ซีพียู ในยุคสิบหกบิต



คำตอบก็คือ ประการแรกที่ว่ากำลังจะพ้นสมัยนั้นเฉพาะในงานประมวลผลข้อมูลดังที่ใช้กันอยู่บนเครื่องแอปเปิ้ลเทียบกับเครื่องไอบีเอ็มพีซี แต่ในงานควบคุมซีพียูขนาด 8 บิต ยังคงมีความหมายอยู่ตราบเท่าที่อุปกรณ์ในการประกอบเป็นระบบคอมพิวเตอร์ขนาด 8 บิต ยังคงถูกกว่าขนาด 16 บิต (คือยังคงผลิตกันอยู่) และงานนั้นไม่สลับซับซ้อนเกินขีดความสามารถของซีพียูขนาด 8 บิต

ประการที่สองการศึกษาทางด้านฮาร์ดแวร์นั้นเน้นหนัก ในทางที่ให้ผู้เข้ารับการศึกษาสามารถออกแบบระบบคอมพิวเตอร์ได้ด้วยตนเอง แต่การที่จะจัดระบบคอมพิวเตอร์ขึ้นมาใหม่นั้นจำเป็นจะต้องมีเครื่องคอมพิวเตอร์ที่สมบูรณ์แล้วเป็นเครื่องแม่ ในยุคของเครื่อง 8 บิต เราสามารถใช้เครื่องที่เรียกว่า ซิงเกิลบอร์ด (Single Board) ดำเนินการได้ เพราะชุดคำสั่งของซีพียูขนาด 8 บิต สามารถจะทำแอสเซมเบลอร์ด้วยมือ (Hand assembler) และป้อนเข้าเครื่องในรูปแบบของเลขฐานสิบหกได้ ซึ่งเครื่องระดับนี้จะมีราคาประมาณ 3,000 บาท อยู่ในวิสัยที่สถาบันการศึกษาจะจัดซื้อหาได้

แต่สำหรับซีพียูขนาด 16 บิต การทำแอสเซมเบลอร์ด้วยมือเป็นเรื่องที่ทำได้ยากมากจนอาจกล่าวว่าทำไม่ได้ (แต่ทั้งนี้ผมก็ต้องขอกล่าวไว้ด้วยว่า ท่านที่อยู่ในยุคแรก ๆ ของคอมพิวเตอร์โดยเฉพาะก่อนที่ไมโครคอมพิวเตอร์จะระบาดท่านก็ทำกัน แต่ทำกันในระดับผู้ชำนาญ) เมื่อทำแอสเซมเบลอร์ด้วยมือไม่ได้ก็ต้องใช้โปรแกรมแอสเซมเบลอร์ ซึ่งจะต้องใช้จอภาพและคีย์บอร์ดแบบพิมพ์ดีด จึงเป็นไปไม่ได้ที่ราคาเครื่องแม่จะอยู่ในเกณฑ์ 3,000 บาท

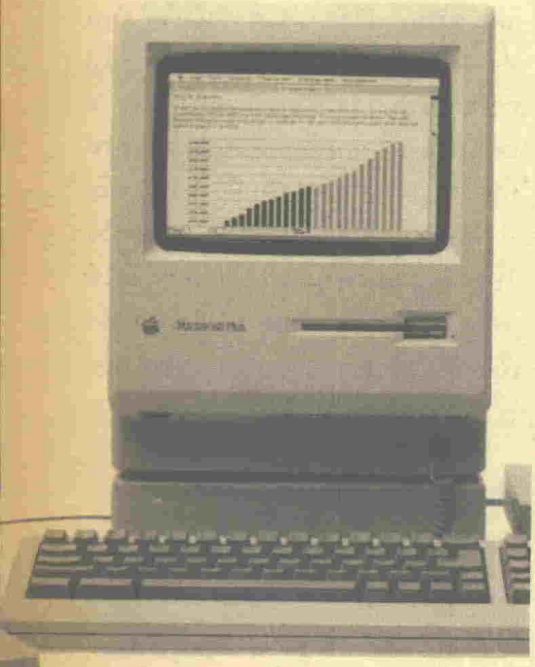
อนาคต

ผลสะท้อนของเรื่องนี้คือ ซิงเกิลบอร์ดขนาด 16 บิต โดยเฉพาะที่ผลิตใน

ประเทศไม่มีออกสู่ตลาดเลย ผู้ที่มีใจรักทางด้านฮาร์ดแวร์ยังคงมีอยู่ แต่อัตราเพิ่มของสมาชิกใหม่น้อยมาก ประหนึ่งว่าทุกคนจะมุ่งไปสู่เรื่องที่เรียนรู้ได้ง่ายกว่าโก้กว่า สังเกตได้จากหน้าโฆษณาย้อนกลับไปเพียง 2 ปี จะมีรายการไอซีพวกซีพียู ชิพซัพพอร์ต และหน่วยความจำ โฆษณากันสะพรั่ง แต่ในปัจจุบันเช่น ในวารสารเซมิคอนดักเตอร์อิเลคทรอนิคส์ ฉบับเดือนเมษายน 2530 มีเพียงรายการเดียว จนผมอยากจะกล่าวว่ายุค 16 บิต ยังเดินทางมาไม่ถึงเมืองไทย (และอาจจะไม่มาถึงเลย)

ในเรื่องของอนาคต ดังที่ทราบกันว่าสิ่งใดที่พ้นสมัยราคาจะแพงขึ้นเพราะเลิกผลิต เราจะปักหลักกันที่ซีพียู 8 บิต ต่อไปคงเป็นไปไม่ได้ วิทยาการก็รุดหน้าไปอย่างรวดเร็ว การติดตามวิทยาการก้าวหน้าทำได้ 2 ลักษณะ คือ ก้าวกระโดด หรือตามขั้นตอน หากต้องการก้าวกระโดดสำหรับกรณีนี้คือเลือกซีพียูขนาด 32 บิต เช่น 80386 หรือ 68030 มาศึกษาและใช้งาน ถ้าต้องการให้เป็นไปตามขั้นตอนก็เลือกซีพียูขนาด 16 บิต มาศึกษาใช้งาน

โดยส่วนตัวของผมนิยมการก้าวกระโดดมากกว่า และถ้าให้ฝันเฟื่องก็



หมายถึงเราจะออกแบบไมโครคอมพิวเตอร์ขนาด 32 บิต ออกขายในตลาดโลก แต่ก็ติดขัดที่ขาดสิ่งอำนวยความสะดวกสนับสนุน ยกตัวอย่างง่าย ๆ คือ การเดินสายจากขาซีพียูออกมาใช้งาน จึงต้องลดระดับมาที่ซีพียูขนาด 16 บิตก่อน ซึ่งสามารถนำมาใช้ในงานควบคุมได้กว้างขวางกว่าขนาด 8 บิต และที่ท้าทายคือเราน่าจะมีเครื่องไมโครคอมพิวเตอร์ที่ออกแบบมาให้เหมาะกับงานประมวลผลภาษาไทยโดยตรง

ซีพียูที่จะนำมากล่าวถึง

ซีพียูขนาด 16 บิต ที่รู้จักกันกว้างขวางคือ 8086 ของอินเทล 68000 ของโมโตโรลา และ Z8000 ของไซล็อก แต่ Z8000 ตกอับไปอย่างรวดเร็วเพราะการออกแบบที่ผิดพลาด ถึงแม้ต่อมาจะได้ปรับปรุงสิ่งที่ผิดก็เสื่อมความนิยมเสียแล้ว ผมจึงจะไม่นำมากล่าวถึง คงเหลือ 8086 กับ 68000 ดังนั้นเรื่องราวซีพียูในยุคสิบหกบิตนี้จึงจะกล่าวถึงซีพียู 2 เบอร์นี้เท่านั้น

ในการกล่าวถึงเรื่องราวของซีพียูเพื่อจะนำมาศึกษาโดยละเอียดต่อไปและเพื่อการเปรียบเทียบกันนั้น ผมเห็นว่าเฉพาะเรื่องของรีจิสเตอร์ ชุดคำสั่ง และแบบของการอ้างแอดเดรสก็นับว่าเพียงพอแล้ว แต่จะนำเรื่องอื่นมากล่าวประกอบด้วย

สถาปัตยกรรมสองแบบ

ก่อนที่จะกล่าวถึงซีพียูทั้งสองเบอร์นี้ ผมเห็นว่าควรจะท้าวความไปถึงซีพียูขนาด 8 บิต อันเป็นจุดกำเนิดของซีพียูทั้งสองนี้ก่อน เป็นที่ทราบกันว่าอินเทลเป็นผู้ริเริ่มยุคของไมโครโปรเซสเซอร์ บริษัทอื่นน่าจะเป็นผู้ตามแต่ปรากฏว่าสถาปัตยกรรมของอินเทลกับโมโตโรลาแตกต่างกัน และแตกต่างกันชนิดว่าแยกออกเป็น 2 ตระกูลให้บริษัทอื่นเจริญรอยตาม กล่าวคือ

ในตระกูลของอินเทลเช่น 8080 ใช้รีจิสเตอร์ชี้ข้อมูล เช่นต้องการโหลดค่าจากแอดเดรส 2000H ให้แก่แอคคิวมูเลเตอร์ จะทำโดยให้ค่า 2000H แก่รีจิสเตอร์ HL ก่อน แล้วใช้คำสั่ง MOV A,M (M หมายถึงหน่วยความจำที่ HL ชี้อยู่) ทำให้มีการกล่าว(หา) กันว่า 8080 เป็นซีพียูในแบบเครื่องคิดเลข (Calculator Type)

ส่วนในตระกูลของโมโตโรลาใช้หน่วยความจำชี้ข้อมูล เช่น ให้หน่วยความจำที่ $0050 และ $0051 มีค่าเป็น $2000

และให้รีจิสเตอร์ X มีค่าเป็นศูนย์ เมื่อใช้คำสั่ง LDA ($50), X จะเป็นการโหลดค่าจากหน่วยความจำที่แอดเดรส $2000 ให้แก่แอคคิวมูลเลเตอร์ A ทำให้ถือกันว่าซีพียูในตระกูลโมโตโรลาเป็นแบบมินิคอมพิวเตอร์ (Minicomputer Type)

การที่กล่าวว่าเป็นแบบเครื่องคิดเลขหรือแบบมินิคอมพิวเตอร์ไม่ใช่เรื่องสำคัญ จุดสำคัญคือโดยที่สถาปัตยกรรมของอินเทลใช้รีจิสเตอร์เป็นหลักจึงต้องมีรีจิสเตอร์เป็นจำนวนมาก แต่ของโมโตโรลาใช้หน่วยความจำแทนดังนั้นจึงมีรีจิสเตอร์น้อยมากคือมีแต่แอคคิวมูเลเตอร์กับอินเด็กซ์เท่านั้น (ไม่นับโปรแกรมเคาน์เตอร์กับสแต็คพอยน์เตอร์)

แต่พอมาถึงยุค 16 บิต สถาปัตยกรรมของซีพียูทั้งสองตระกูลนี้กลับไขว้เข้าหากัน คือของอินเทลได้นำอินเด็กซ์มาใช้ ส่วนของโมโตโรลาได้เลิกใช้หน่วยความจำชี้ข้อมูลหันมาใช้รีจิสเตอร์แทน และกลับมีรีจิสเตอร์มากกว่าของอินเทล

สิ่งที่พัฒนา

การก้าวขึ้นสู่ยุค 16 บิตนี้ ไม่เพียงแต่จำนวนบิตต่อเวอร์ดเท่านั้นที่เพิ่มขึ้น แต่ขีดความสามารถของซีพียูทั้งสองนี้ก็เพิ่มขึ้นด้วยเช่นกัน ซึ่งต่างก็เน้นหนักในการสนับสนุนการเขียนโปรแกรมระบบจัดการและโปรแกรมตัวแปรภาษา กล่าวคือ

โปรแกรมระบบจัดการ

อินเทลได้จัดอินเตอร์รัพต์เวคเตอร์ให้แก่ 8086 ไว้ถึง 256 เวคเตอร์ โดยให้ใช้ร่วมกันทั้งในด้านฮาร์ดแวร์อินเตอร์รัพต์ คือ การอินเตอร์รัพต์ที่ขานี้ของซีพียู และซอฟต์แวร์อินเตอร์รัพต์ คือ การอินเตอร์รัพต์ด้วยคำสั่ง INT n โดยที่ n หมายถึงหมายเลขอินเตอร์รัพต์ คือ หมายเลขเวคเตอร์

วิธีการนี้ทำให้สะดวกในการเขียนและเปลี่ยนแปลงแก้ไขโปรแกรมระบบจัดการ เพราะถึงจะเปลี่ยนแปลงโปรแกรมระบบจัดการอย่างไร ก็จะไม่กระเทือนไปถึงโปรแกรมที่ใช้บริการของระบบจัดการ เช่น โปรแกรมตัวแปรภาษา หรือโปรแกรมใช้งาน ทั้งนี้เพราะถ้าแอดเดรสของโปรแกรมย่อยในระบบจัดการเปลี่ยนไปก็เพียงให้ค่าแอดเดรสใหม่นี้ในเวคเตอร์เดิม ซึ่งเป็นหน้าที่ของระบบจัดการ

สำหรับ 68000 ของโมโตโรลาก็ได้จัดอินเตอร์รัพต์เวคเตอร์ไว้ถึง 256 เวคเตอร์เช่นกัน และมีคำสั่ง TRAP เพื่อให้ใช้เรียกโปรแกรมย่อยในระบบจัดการได้เช่นกัน แต่ที่ของโมโตโรลาเหนือกว่าของอินเทลคือ แยกโหมดการทำงานของซีพียูออกเป็น 2 โหมด คือ Supervisor และ User

โหมด Supervisor มีความประสงค์จะให้ใช้กับโปรแกรมระบบจัดการ ส่วนโหมด User สำหรับโปรแกรมใช้งาน การแยกโหมดกันนี้มีความสัมพันธ์กับทางฮาร์ดแวร์ด้วย เพราะจะแยกหน่วย

ความจำออกจากกันอย่างเด็ดขาด ดังนั้นหากมีข้อผิดพลาดเกิดขึ้นในโปรแกรมใช้งาน จะไม่มีผลถึงโปรแกรมระบบจัดการ คือไม่ทำให้เกิดอาการที่เรียกว่าค้าง (เครื่องไม่ทำงาน ดังที่ปรากฏในเครื่อง 8 บิต และเครื่องที่ใช้ 8086/8088)

นอกจากนี้โมโตโรลายังได้จัดสิ่งอำนวยความสะดวกไว้สำหรับการรันโปรแกรมขนาน (concurrent) ซึ่งผมเห็นว่าไม่ใช่เรื่องสำคัญนัก เพราะสามารถเขียนระบบจัดการให้รันโปรแกรมขนานกับ 8086 ได้เช่นกัน

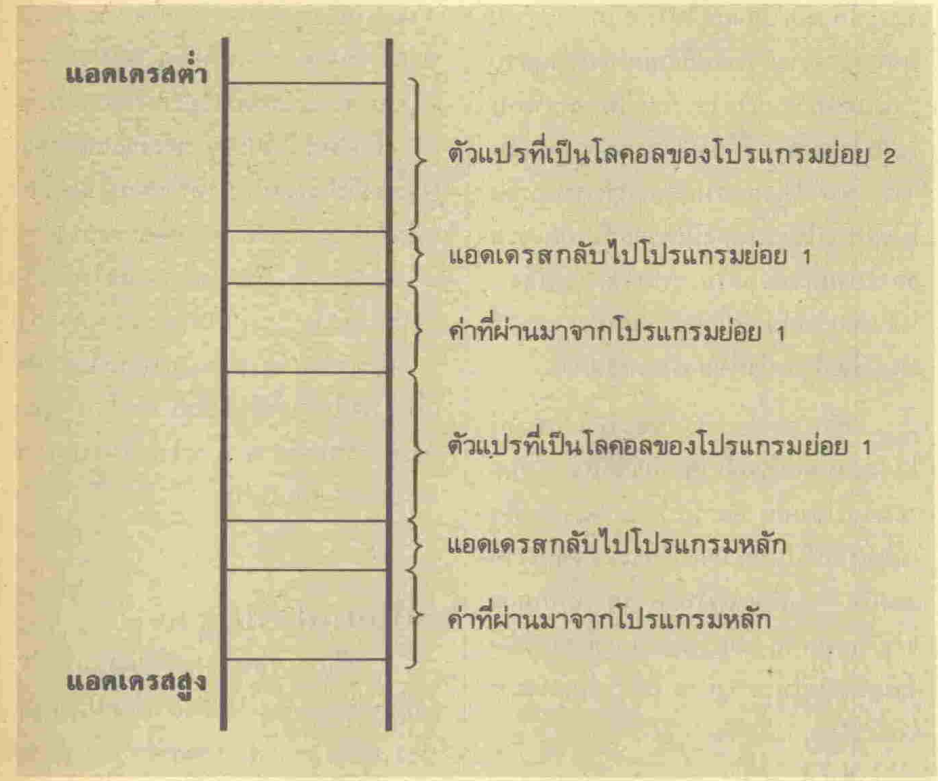
สนับสนุนภาษาสูง

ซีพียูรุ่นก่อนรวมทั้งของคอมพิวเตอร์ขนาดเมนเฟรมด้วยได้ออกแบบมาเพียงเพื่อ ให้สามารถติดต่อกับอุปกรณ์รอบข้างได้ทางส่วนเชื่อมต่อ (Peripheral interface) ให้สามารถอ่านและเขียนกับหน่วยความจำได้ กับให้สามารถดำเนินกรรมวิธีทางคณิตศาสตร์และตรรกได้เท่านั้น ไม่ได้คำนึงถึงการอำนวยความสะดวกให้แก่การเขียนโปรแกรมตัวแปรภาษา (Compiler หรือ Interpreter)

แต่ในซีพียูรุ่นใหม่นี้ได้คำนึงถึงเรื่องนี้ด้วยเช่นในเรื่องการเขียนโปรแกรมเป็นโครงสร้าง (Structure programming) การผ่านค่า และโครงสร้างข้อมูล (Data structure) ซึ่งมีอยู่ในซีพียูทั้งสองเบอร์นี้

การสนับสนุนในด้านการเขียนโปรแกรมเป็นโครงสร้างหมายถึงทำให้การเขียนโปรแกรมตัวแปรภาษาที่เป็นโครงสร้างเช่นภาษาปาสคาลและซีทำได้ง่ายขึ้น เช่น มีคำสั่งให้ทำงานเป็นวงรอบ และจากบทความชุดภาษาเครื่องสำหรับภาษาสูงจะเห็นว่า เราสามารถแปลงร่างโปรแกรมที่เขียนเลียนแบบโปรแกรมในภาษาปาสคาล เป็นภาษาแอสเซมเบลอร์ อย่างที่เรียกได้ว่าเกือบจะเป็นบรรทัดต่อบรรทัด

ในเรื่องของการผ่านค่า โปรแกรมตัวแปรภาษาส่วนมากจะผ่านค่าทางสแต็ค



รูปที่ 1 แสดงข้อมูลในสแต็คเมื่อมีการผ่านค่าและตัวแปรที่เป็นโลคอลของโปรแกรมย่อย โดยสมมติว่าโปรแกรมหลักเรียกใช้โปรแกรมย่อย 1 โดยมีการผ่านค่าไปทางสแต็ค โปรแกรมย่อย 1 มีตัวแปรที่เป็นโลคอลของตนเอง และเรียกใช้โปรแกรมย่อย 2 อีกต่อหนึ่งโดยมีการผ่านค่าไปให้และโปรแกรมย่อย 2 ก็มีตัวแปรที่เป็นโลคอลด้วยเหมือนกัน

และผมขอหมายรวมถึงตัวแปรที่เป็นโลคอล (local variable) เช่น ในภาษาปาสคาลและซีด้วย เพราะโปรแกรมตัวแปลภาษาจะกำหนดให้ตัวแปรเหล่านี้อยู่ในสแต็ค ดังแสดงในรูปที่ 1 ใน 8086 กำหนดให้เราใช้รีจิสเตอร์ BP ร่วมกับค่า displacement ชี้ข้อมูลเหล่านี้ได้โดยสะดวก

สำหรับ 68000 สามารถทำได้เช่นกัน โดยใช้ Address register ร่วมกับค่า displacement และดีกว่าด้วยคือ มีคำสั่ง LINK กับ UNLINK ให้เราดำเนินการกับสแต็คได้โดยสะดวกกว่า

การดำเนินการกับข้อมูลโดยเฉพาะในเรื่องการเคลื่อนย้ายข้อมูลในอะเรย์สำหรับเครื่อง 8 บิต จะทำได้กับข้อมูลขนาด 8 บิตเท่านั้น (เว้น 6809 สามารถดำเนินการกับข้อมูล 16 บิตได้ด้วย) สำหรับซีพียู 16 บิตสองตัวนี้ สามารถดำเนินการได้กับข้อมูล 8 บิต และ 16 บิต แต่ 68000 สามารถดำเนินการกับข้อมูลขนาด 32 บิตได้ด้วย (เรื่องนี้ไม่ใช่วิทยาการก้าวหน้าเพราะซีพียูขนาด 16 บิต ย่อมดำเนินกรรมวิธีครั้งละ 16 บิต ที่นำมากล่าวเพื่อให้สมบูรณ์เท่านั้นครับ)

แต่ในเรื่องของอะเรย์แท้ ๆ ซีพียูทั้งสองเบอร์นี้ มีแบบของการอ้างแอดเดรสใกล้เคียงกับอะเรย์ในภาษาสูง เช่น ในภาษาปาสคาลหากเขียนว่า

num : = numArray[I]

โดยหมายถึงให้นำค่าเลขจำนวนเต็มใน numArray ตรงช่อง I มาให้แก่ตัวแปร num จะเขียนเป็นภาษาแอสเซมบลีของ 8086 ได้เป็น

```
MOV AX, numArray. [SI]
MOV num,AX
```

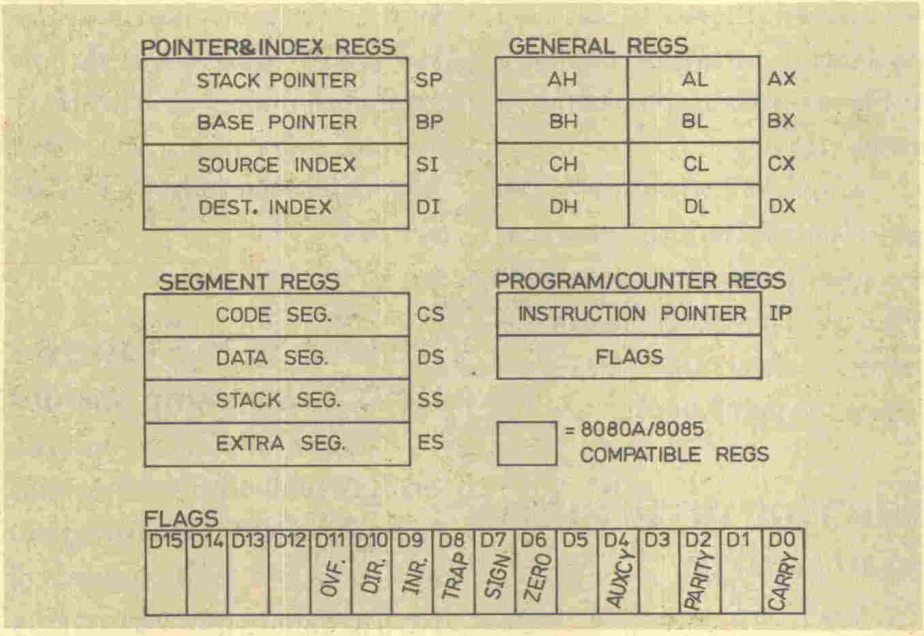
การอ้างแอดเดรสแบบ numArray.[SI] ซึ่งก็คือ [numArray+SI] นั่นเอง โดยที่ numArray เป็นค่าแอดเดรสของต้นอะเรย์ numArray ส่วน SI เป็นอินเด็กซ์จากต้นอะเรย์ แบบของการอ้างแอดเดรสแบบนี้ดูว่าน่าจะจัดให้มีได้ไม่ยากนัก แต่ก็ไม่มีในซีพียูรุ่นเก่า

สำหรับข้อมูลแบบเรคอร์ดซึ่งดูสลับซับซ้อนกว่า กลับสามารถทำได้ในซีพียูรุ่น 8 บิต โดยใช้แบบของการอ้างแอดเดรสในแบบ Direct addressing mode แต่ดูไม่เป็นเรคอร์ดในแบบภาษาสูง เพราะดำเนินกรรมวิธีได้ครั้งละ 8 บิต และไม่มีแบบการอ้างแอดเดรสกับอะเรย์ดังกล่าวมา พอมาถึงยุค 16 บิต ซึ่งดำเนินการในเรื่องดังกล่าวมาได้ จึงใกล้เคียงกับเรคอร์ดในภาษาสูงมาก ดังการกำหนดและการใช้ตัวแปรแบบ Structure ใน MASM ซึ่งเป็นแอสเซมเบลอร์ของ 8086 สิ่งเหล่านี้ทำให้โปรแกรมตัวแปรภาษาดำเนินการกับข้อมูลแบบต่าง ๆ ได้สะดวกขึ้น

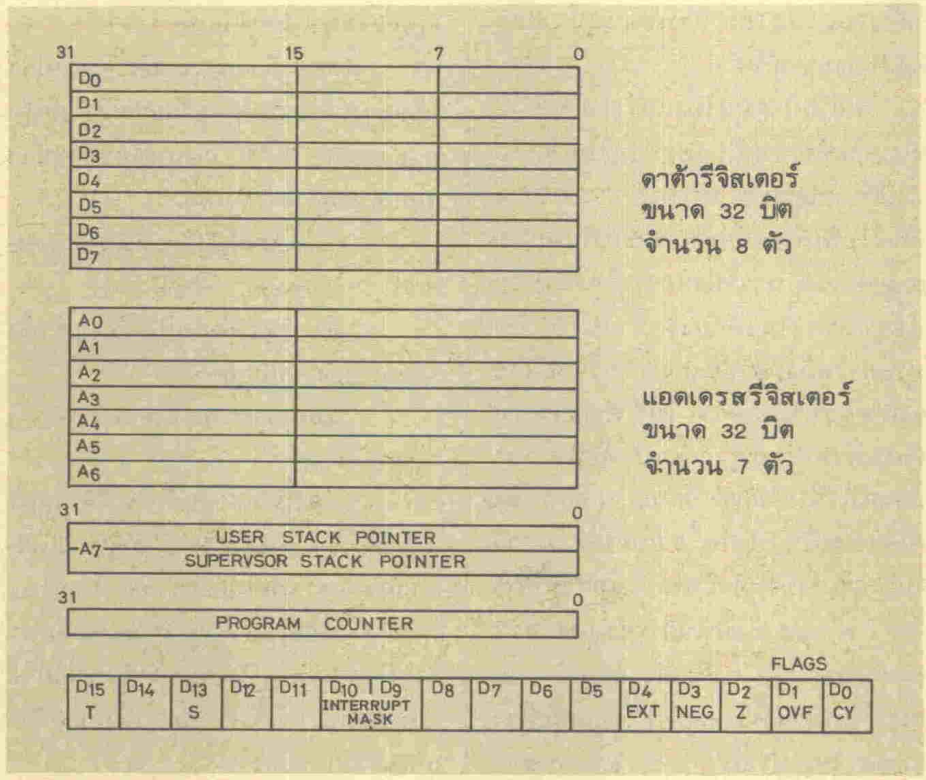
รีจิสเตอร์

โดยปกติซีพียูขนาดใดก็จะมีแอคคิวมูเลเตอร์และรีจิสเตอร์ใช้งานทั่วไป (general purpose register) ขนาดเดียวกัน เช่น 6800 มีแต่แอคคิวมูเลเตอร์ขนาด 8 บิตเท่านั้น แต่ถ้าซีพียูเบอร์ใดต้องการเพิ่มสมรรถภาพของตนให้เหนือกว่าเบอร์อื่นก็จะเพิ่มขนาดขึ้นเป็น 2 เท่า เช่น 8080 มี BC,DE และ HL เป็นรีจิสเตอร์ขนาด 16 บิต โดยใช้ HL เป็นแอคคิวมูเลเตอร์ขนาด 16 บิตด้วย

เป็นที่น่าประหลาดใจว่าในรุ่น 8 บิตอินเทลสร้าง 8080 ให้มีส่วนที่เป็น 16 บิต แต่พอมาถึงยุค 16 บิตกลับสร้าง 8086 ให้มีแต่รีจิสเตอร์ขนาด 16 บิตเท่านั้น ตรงข้ามกับโมโตโรลาซึ่งสร้าง 6800 ให้เป็น



รูปที่ 2 แสดงภาพรีจิสเตอร์ของ 8086 ที่เป็นรีจีสเตอร์ใช้งานทั่วไปคือ AX, BX, CX และ DX ที่เป็นอินเด็กซ์ คือ SI และ DI สำหรับ BX มีหน้าที่พิเศษคือเป็น base register ใช้งานร่วมกับอินเด็กซ์ ส่วน BP ทำหน้าที่เป็น base register เช่นกันสำหรับข้อมูลในสแต็ค



รูปที่ 3 รีจิสเตอร์ของ 68000 ที่เป็นรีจิสเตอร์ใช้งานทั่วไปคือ D_0 ถึง D_7 เป็น data register สำหรับพักข้อมูลในการดำเนินกรรมวิธีข้อมูล และ A_0 ถึง A_6 เป็น address register สำหรับชี้ข้อมูล สามารถใช้ทั้ง data register และ address register เป็นอินเด็กซ์ได้

ซีพียูขนาด 8 บิตแท้ ๆ แต่พอมาถึง 68000 กลับกลายเป็นขนาด 32 บิตอยู่ภายในขอให้ดูในรูปที่ 2 และ 3 ประกอบด้วย

เรื่องนี้ทำให้มีข้อพิจารณาว่าถ้าเรียก 8088 ซึ่งเป็น 8086 ที่มีขาดาต้าเพียง 8 ขา ว่าเป็นซีพียูขนาด 16 บิต ก็ต้องเรียก 68000 ว่าเป็นซีพียูขนาด 32 บิตด้วย เพราะสถาปัตยกรรมและโครงสร้างภายในของ 68000 เป็น 32 บิต เช่นเดียวกับที่ 8088 เป็น 16 บิต

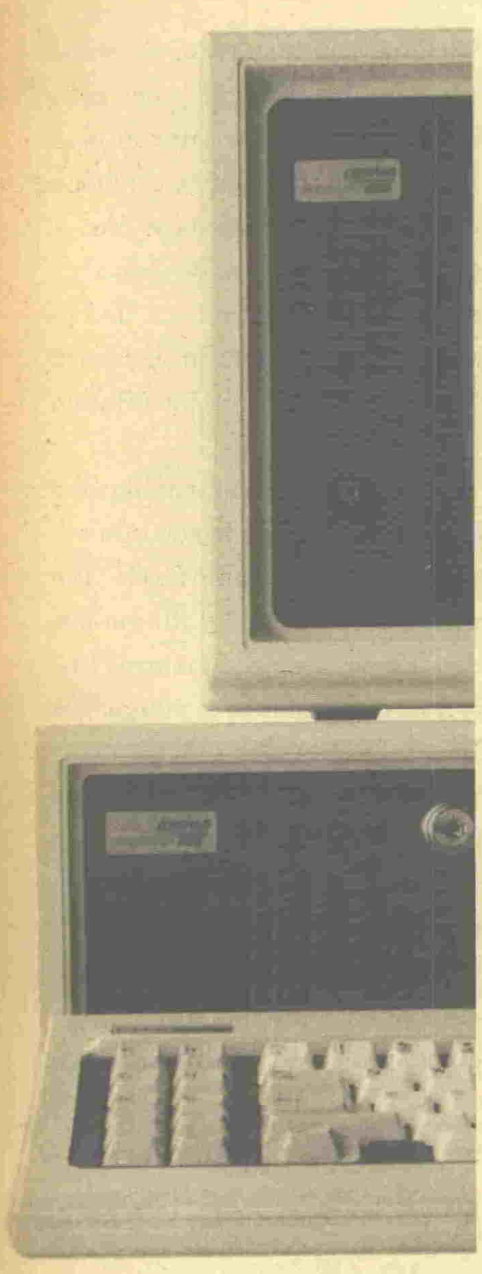
จากการกำหนดรีจิสเตอร์ในรูปที่ 2 และที่ 3 จะเห็นว่าทั้งอินเทลและโมโตโรลาต่างนำความคิดเรื่องอินเด็กซ์มาใช้ โดยที่ใน 8086 มี SI และ DI เป็นอินเด็กซ์ ส่วน 68000 สามารถใช้แอดเดรสรีจิสเตอร์และดาต้ารีจิสเตอร์ทุกตัวเป็นอินเด็กซ์ได้

สถาปัตยกรรมที่แตกต่างกันอีกเรื่องหนึ่งคือ อินเทลยังคงแนวความคิดเดิมของตนที่ให้รีจิสเตอร์แต่ละตัวมีหน้าที่พิเศษ ส่วนโมโตโรลาให้รีจิสเตอร์ทุกตัวทำงานได้เหมือนกันในพวกของมัน คือ ดาต้ารีจิสเตอร์กับแอดเดรสรีจิสเตอร์ ซึ่งถือว่าซีพียูในยุคใหม่ควรให้รีจิสเตอร์ทุกตัวทำงานได้เหมือนกัน

ชุดคำสั่ง

เมื่อ Z80 ออกมาใหม่บุคคลในวงการไมโครคอมพิวเตอร์ต่างก็พากันตื่นเต้นในชุดคำสั่งอันมากมายถึง 158 คำสั่ง และถือกันว่าเป็นความสามารถของซีพียู แต่ปัจจุบันถือกันว่าชุดคำสั่งควรมีไม่มากนัก แต่ให้สามารถอ้างแอดเดรสได้มากแบบและอย่างมีประสิทธิภาพ

ทั้งอินเทลและโมโตโรลาต่างก็ถือตามนี้ แต่ถึงกระนั้น 8086 ก็มีชุดคำสั่งถึงประมาณ 100 คำสั่ง ส่วน 68000 มีเพียง 56 คำสั่ง คำสั่งส่วนใหญ่แล้วจะเหมือนกันได้แก่ การเคลื่อนย้ายข้อมูล การคำนวณกรรมวิธีทางตรรก การโรเตต และการชิฟต์ การกระโดดและเรียกโปร-



แกรมย่อย ซึ่งน่าจะมีจำนวนคำสั่งใกล้เคียงกัน เหตุที่ของ 8086 มีมากกว่า เพราะมีคำสั่งเบ็ดเตล็ดมากกว่า และนับจำนวนคำสั่งไม่เหมือนกัน เช่น BCc ของ 68000 นับเป็นหนึ่ง แต่ถ้านับตามแบบของอินเทลจะเป็น 14 คำสั่ง

คำสั่งที่เพิ่มขึ้นในยุค 16 บิต นอกจากคำสั่งที่เกี่ยวกับการอำนวยความสะดวกในการเขียนโปรแกรมตัวแปรภาษาก็ได้แก่คำสั่ง ในการคูณ-หาร และการดำเนินกรรมวิธีกับเลขจำนวนคือเครื่องหมายบวกลบ ซึ่งนับว่าเป็นการพัฒนาไปอีกขั้นตอนหนึ่ง ซีพียูในยุคแรก ๆ นั้นบางตัวแม้แต่คำสั่งในการลบก็ไม่มี มีแต่การบวก หากต้องการลบจะต้องเขียนโปรแกรมการลบ

ความก้าวหน้าทางด้านการคำนวณแท้ ๆ ได้ออกนอกซีพียูไปอยู่บนชิพคำนวณ เช่น 8086 มี 8087 สำหรับ 68000 เท่าที่ผมทราบยังไม่มีชิพคำนวณของตนเอง แต่สามารถใช้ชิพประเภท Signal Processing มาช่วยดำเนินการได้

ขีดความสามารถในการอ้างแอดเดรส

ขีดความสามารถในการอ้างแอดเดรสของซีพียูรุ่นใหม่นับว่าสำคัญมาก ในสมัยก่อนแม้แต่บนเครื่องเมนเฟรมก็มีหน่วยความจำเพียงร้อยกิโลไบต์หรือไม่ถึงเท่านั้น เมื่ออินเทลออก 8080 ให้มีขีดความสามารถในการอ้างแอดเดรสได้ถึง 64 กิโลไบต์ ก็ชื่นชมกันมากและคิดกันว่ามากเกินตัว

แต่ในปัจจุบันโปรแกรมแต่ละตัวมีขนาดโตขึ้นเช่นเป็นร้อยกิโลไบต์ขึ้นไป ทั้งนี้ด้วยเหตุผล 2 ประการ ประการแรกเป็นการเพิ่มขีดความสามารถของโปรแกรมเอง ดังจะเห็นว่าโปรแกรมสำเร็จรูปในปัจจุบันทำงานได้อย่างกว้างขวาง อีกประการหนึ่งคือโปรแกรมสำเร็จรูปและแม้แต่ระบบจัดการ ในปัจจุบันนิยมเขียนด้วยภาษาสูง เพื่อประหยัดเวลาและค่าใช้จ่ายในการเขียนโปรแกรม ซึ่งเมื่อแปลเป็นรหัสเครื่องแล้วจะโตกว่าโปรแกรมที่เขียนด้วยภาษาเครื่องโดยตรง

ความสามารถในการอ้างแอดเดรสของ 8086 นับว่าเป็นรอง 68000 มาก เพราะสามารถอ้างแอดเดรสได้สูงสุดเพียง 1 เมกะไบต์เท่านั้น ส่วน 68000 สามารถอ้างได้ถึง 16 เมกะไบต์ อีกเรื่องหนึ่งคือการอ้างแอดเดรสของ 8086 ไม่สามารถอ้างได้อย่างต่อเนื่องทั้ง 1 เมกะไบต์ (เพราะใน 8086 มีแต่รีจิสเตอร์ขนาด 16 บิต สามารถให้ค่าแอดเดรสได้ในย่าน 64 กิโลไบต์เท่านั้น) ต้องแบ่งออกเป็นเซกเมนต์ละ 64 กิโลไบต์ 16 เซกเมนต์ ก่อความยุ่งยากในการเขียนโปรแกรมขนาดใหญ่กว่า 64 กิโลไบต์พอสมควร

แบบของการอ้างแอดเดรส (Adressing mode)

ดังที่กล่าวแล้วว่าการออกแบบซีพียูในปัจจุบันนิยมให้มีคำสั่งในชุดคำสั่งไม่มากนัก แต่เอาดีกันทางด้านแบบของการอ้างแอดเดรส ให้มีมากแบบอย่างมีประสิทธิภาพและสนับสนุนการเขียนโปรแกรมตัวแปรภาษา ในยุค 8 บิต ซีพียู Z80 ซึ่งถือกันว่าเด่นที่สุดในพวกอินเทลก็มีแบบของการอ้างแอดเดรสเป็นรอง 6809 ของโมโตโรลา และถือกันว่าโมโตโรลาเด่นในเรื่องแบบของการอ้างแอดเดรสมากกว่าอินเทล

แต่พอมาถึงยุค 16 บิต ผมเห็นว่ามีขีดความสามารถพอ ๆ กันและสนับสนุนภาษาสูงเท่า ๆ กัน มีแบบของการอ้างแอดเดรสคล้าย ๆ กันคือ

- ถึงรีจิสเตอร์ (Register addressing)
- ถึงค่าคงตัวที่อยู่ต่อท้ายคำสั่ง (Immediate addressing)
- ถึงหน่วยความจำโดยตรง(Direct addressing)
- ถึงหน่วยความจำผ่านรีจิสเตอร์ + อินเด็กซ์ + ค่าขจัด (Register indirect addressing: with index and displacement) คือจะอ้างถึงทั้งรีจิสเตอร์และอินเด็กซ์รวมกัน หรือตัวใดตัวหนึ่งก็ได้ หากมีแต่ค่าขจัดก็จะกลายเป็นการอ้างแอดเดรสโดยตรงไป

แบบการอ้างแอดเดรสที่เคยมีในยุค 8 บิต แต่หายไปคือการอ้างแอดเดรสในแบบ Indirect addressing โดยนำแบบ Register indirect addressing มาใช้แทน

และที่ผมเคยกล่าวว่าทำให้อ้างแอดเดรสถึงข้อมูลแบบอะเรย์ได้ ความจริงแล้วการอ้างแอดเดรสถึงข้อมูลแบบอะเรย์น่าจะกระทำในแบบ Direct with index แต่เนื่องจากในซีพียูทั้งสองเบอร์นี้ สามารถให้ค่าขจัด (displacement) ขนาด 16 บิตได้ จึงเทียบได้กับว่าเป็น Direct with index ไปในตัว

ซึ่งของ 8086 ต้นอะเรย์จะอยู่ ณ ที่ใดก็ได้ในเซกเมนต์ แต่ของ 68000 เพราะค่าขจัดเป็นเลขเครื่องหมาย หากใช้วิธีนี้ต้นอะเรย์จะต้องอยู่ใน 32 กิโลไบต์แรก ถ้าเกินไปจากนี้ก็ต้องใช้รีจิสเตอร์อีกตัวหนึ่งเป็นฐานโดยให้ชี้ไปที่ต้นอะเรย์

ความเร็ว

สิ่งที่พัฒนาต่อจากซีพียูในยุค 8 บิต อีกเรื่องหนึ่งคือ ในด้านของความเร็ว ได้พัฒนาสารกึ่งตัวนำประเภท NMOS ให้มีความเร็วสูงขึ้นเรียกว่า HMOS และได้ผลผลิตออกมาในหลายระดับความเร็ว เช่นของ 8086 มีรุ่นที่ใช้กับสัญญาณนาฬิกาดังนี้คือ 5, 8 และ 10 MHz ส่วนของ 68000 มีรุ่น 4, 6, 8, 10 และ 12.5 MHz

เรื่องที่แปลกคือ ในตระกูลของโมโตโรลาเดิมใช้สัญญาณนาฬิกาในแบบ 1 clock ต่อ 1 การปฏิบัติ เช่น การอ่านคำสั่งจะใช้สัญญาณนาฬิกา 1 clock เท่านั้น ของอินเทลใช้หลาย clock ต่อหนึ่งการปฏิบัติ เช่น การอ่านคำสั่งใช้ 3 clock แต่พอมาในยุค 16 บิต ของอินเทลคงเดิม ของโมโตโรลากลับใช้หลาย clock ต่อหนึ่งการปฏิบัติโดยที่จำนวน clock มากกว่าของอินเทล

สะพานเชื่อม

ทั้งอินเทลและโมโตโรลาต่างก็เห็นว่าการที่จะให้ผู้ใช้ที่เคยชินกับซีพียูขนาด 8 บิต กระโดดมาใช้ซีพียูขนาด 16 บิตโดยตรง คงจะทำได้ไม่ง่ายนัก จึงได้ออกรุ่นที่มีขาดาต้า 8 บิต ออกมาเป็นสะพานเชื่อมระหว่างยุค 8 บิต กับ 16 บิต ของ 8086 คือ 8088 ส่วนของ 68000 คือ 68008

ก่อนจบ

ในความคิดของผมยุค 8 บิต สำหรับงานประมวลผลข้อมูล (data processing) ได้ผ่านไปแล้ว คงอยู่แต่ในงานควบคุม (control) แต่จะอยู่ได้นานเท่าใด

เราจำเป็นต้องต้อนรับยุค 16 บิต ซึ่งผมเห็นว่าหากเป็นงานควบคุม 8088 และต่อไป 8086 ดูจะเหมาะกว่าเพราะจัดระบบได้เล็กกว่าและเราก็มีเครื่องแม่เครื่องไอบีเอ็ม- พีซีอยู่เป็นจำนวนมากแล้ว แต่สำหรับงานประมวลผลข้อมูลควรเป็น 68000 เพราะมีสมรรถนะเหนือกว่า 8086

บางท่านอาจสงสัยว่าหาก 68000 ดีกว่าจริง ทำไมไอบีเอ็มถึงใช้ 8088 ในเครื่องรุ่นพีซีของตน ผมหาเหตุผลให้ไม่ได้เลย นอกจากจะทราบมาอย่างไม่มีการยืนยันว่าไอบีเอ็มมีหุ้นอยู่ในโมโตโรลา แต่ที่แน่คือ ถ้าไอบีเอ็มไม่ใช้ 8088 ซีพียูเบอร์นี้คือ 8086 และ 8088 ก็คงไปไม่รอด จึงได้แต่เสียดายว่าโมโตโรลาออก 6809 ได้เยี่ยมที่สุดในยุค 8 บิต และ 68000 ได้เยี่ยมที่สุดในยุค 16 บิตรุ่นแรก แต่กลับไม่อยู่ในความนิยม